UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/587,057	07/21/2006	Syouzou Niwata	U 016403-2	6940

140 7590 07/31/2008
LADAS & PARRY LLP
26 WEST 61ST STREET
NEW YORK, NY 10023

EXAMINER	
RUIZ, ANGELICA	
ART UNIT	PAPER NUMBER
2169	
MAIL DATE	DELIVERY MODE
07/31/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/587,057	**Applicant(s)** NIWATA ET AL.	
	Examiner ANGELICA RUIZ	**Art Unit** 2169	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 28 May 2008.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-14 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-14 is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. The Action is responsive to Applicant's amendment, filed on May 8, 2008, in which claims 1-14 are pending for further examination.

It is acknowledged that As a result of the amendment, Claims 1, 2, 10, 11, 12, 13, have

Specification

4. The abstract of the disclosure is objected to because the content of it is not proper format the abstract is a concise statement of the disclosure of the patent. Correction is required. See MPEP § 608.01(b)

Response to Arguments

5. Applicant's arguments with respect to claims 1-14 have been considered but are moot in view of the new grounds of rejection necessitated by Applicant's amendment.

Applicant argues in substance in FACT 1, page 11, that Bauer's system provide at least two computers which operates under different OSs, which is one of the features of the mentioned prior art, but in the claimed language of the present application claim 1 recites "the computer system under operation of said OS program" which is not being specific about using a single computer, the mentioned computer system according to the specification (Fig. 7), shows a "network" which can be more that one "computer system" and according to: (Par [0095], "where another computer system is logged into via a network"; Bauer's prior art covers the mentioned limitations. It is suggested that the claim language be amended to clarify that the claimed invention refers to a single unit computer system comprising only one computer with only one operating system, and provide the citation of the proper paragraphs that enable such claim.

For at least these reasons, the Examiner maintains the prior art rejections.

Claim Objections

6. In view of compliance with the content of MPEP 608.01(m), The Examiner withdraws the objections to Claims 1-14.

Claim Rejections - 35 USC § 101

7. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior office action.

8. In view of the amendments to the claims, the Examiner withdraws all pending rejections under 35 U.S.C. 101.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-3, 6, 7, 9, 10, and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Bauer et al (US Patent No. 5,627,996)**, in view of **Clark et al. (2005/0050054 A1)** and **Luke et al. (US Publication No. 2002/0116402).**

As per Claim 1, Bauer discloses:

A computer system comprising: a memory (10), whereon data are spread; a program execution unit (20), applying processes, based on a predetermined program including an OS program, to the data spread on the memory; a file storage unit (50), storing files formed of predetermined data;

(Abstract, “A **computer-based file system** enables user access to any of a plurality of previously-**stored data files**, each file being identified by at least two file names formatted using different file name formats…”) and (Col. 4, lines 54-60, “Some of the system calls for controlling **processes include the following**: fork (create a new process), exec (overlay the image of a **program** onto the running process), exit (finish **executing a process**), wait (synchronize **process execution** with the exit of a previously forked process), and (control the size of **memory** allocated to a process)”) and (Col. 4, lines 14-31, “The file system driver 132 manages files, **allocating file space**, controlling access to files, and retrieving data for users. **Processes interact with the file system driver...)** and (Col. 2, lines 6-7, “client operating systems”)

- ***a storage processing unit (30), storing the data, spread on the memory, into the file storage unit as a file having an instructed, predetermined filename; a filename presentation unit (60), presenting, to a user, filenames corresponding to the respective files stored in the file storage unit; an spread processing unit (40), spreading, on the memory,***

(Col. 4, lines 15-46, “The system call interface 131 represents the border between **user level 120 (user programs 121 and program libraries 122)** … file system driver 132 and those that interact with the **process control** … storage devices. …”) and (Abstract,

"…two file names formatted using different file name formats…") and (Col. 3, lines 16-23, "The network includes a server computer … illustratively, **provides the client computers** 102, 103 **shared access to data stored on hard disk** 180.") and (Col. 7, lines 14-30, "…either user space or kernel space; file mode is the open mode…") and (Col. 4, lines 14-31, "The file system driver 132 manages files, **allocating file space**, controlling access to files, and retrieving data for users. **Processes interact with the file system driver...).**

- data inside a file corresponding to a specific filename selected by the user from among the filenames presented by the filename presentation unit; and a user interface unit (70), serving an interface function with respect to the user;

(Abstract, "A computer-based file system enables user access to any of a plurality of previously-stored data files, **each file being identified by at least two file names formatted using different file name formats**. The system receives a user request including a purported file name having…") and (Col. 3, lines 62-64, "The user level 120 **interfaces to clients** (hereinafter users) 102...") and (Col. 6, lines 1-5, "VFS provides a **file-system-type independent interface to programs and users**").

- the computer system further comprising: a user recognition unit (90), recognizing a user who is logged in the computer system under operation of said OS at the present time; a storage control unit (35), which, when the storage processing unit (30) is storing a file, inputs a filename-for-user from the user who is logged in,

(Col. 3, lines 57-60, "…computer-based file server, operates under control of a UNIX operating system 105...) and (Col. 11, lines 12-15, "More generally in response to a **client computer user input, a user program may specify via a system call or other mechanism the file name format for subsequent purported file names entered by the user.**") and (Col. 8, lines 19-22, "In step 501 the requester's **execute permission** in the current directory is checked in the standard way. If permission does not exist, an access error message is returned to the user in step 502").

- executes a filename conversion process of converting the filename-for-user to a filename-for-storage based on a predetermined algorithm, and provides to the storage processing unit <u>an instruction to perform storage using the filename-for-storage</u>;

(Col. 9, lines 44-52, "One might implement "**format-dependent matching**" **algorithms for each type of client operating system** that a file server supports. These format-dependent matching algorithms would enable "on-the-fly" **conversions**…").

- a correspondence information storage unit (80), which, when the filename conversion process is carried out by the storage control unit (35), stores information, indicating a correspondence between the filename-for-user and the filename-for-storage, as filename correspondence information for the user who is logged-in;

(Col. 9, lines 36-43, "With reference to FIGS. 7 and 8, we describe a lookup strategy which uses a predetermined algorithm (**file name mapping function**) to determine the DOS **base name equivalents** of the UNIX file names 812, 814 **and 816 of directory**

800. FIG. 9 illustrates a list of **typical algorithms that might be used to map the default (standard) file names** (e.g., 811, 813, 815) to the selected alternate name type.”).

- ***a presentation control unit (65), which, when the filename presentation unit (60) is performing a presentation of filenames*** *under operation of said OS*, ***references the filename correspondence***

(Col. 3, lines 57-60, “…computer-based file server, operates under control of a UNIX operating system 105...) and (Col. 4 and 5, lines 61-67 and 1, respectively, “With joint reference to FIGS. 1, 2 and 3 we describe an overview of a file system. Every file is named by one or more path names, 310. A path name, as shown in 310, includes file names (e.g., home) separated by delimiters (/). The internal representation of a file is given by an inode, 200, which contains a description of the disk layout of the file data and other information such as the file owner, access permissions, and access times.”), the "presentation control unit” being the “internal representation”.

- ***information for the user*** *who is logged in* ***from inside the correspondence information storage unit (80) and provides an instruction to present the filename-for-user in place of the filename-for-storage based on the filename correspondence information referenced;***

(Col. 2, lines 36-49, “a computer-based file system enables access to any of a plurality of previously-**stored data files stored in a storing means**…The system then **accesses the storage device** and checks file names therein which utilize the identified file name format to locate a data file having a file name which is the same as said base name.”).

- *information referenced so that said filename presentation unit presents the filename-for-user instead of the filename-for storage if a correspondence of the filename-for-storage is indicated in the referenced filename correspondence information, whereas said file name presentation unit presents the filename-for storage as it is if no correspondence of the filename-for-storage is indicated in the referenced filename correspondence information.; and*

(Col. 6, lines 12-16, “The present invention permits a programmatic or human user of the file system (hereinafter collectively referred to as a user) to access **files using aliases or alternate names** that may be different from the standard file name of the file stored in the directory structure.”) and (Col. 8, lines 14-56, "With reference to FIG. 5 we now describe the present invention, as illustratively embodied, as a **file system-specific lookup feature**. We describe the processing of the **file name** "home" of our example path … returned to the client.”).

- ***a spread control unit (45), which, when the spread processing unit (40) spreads data, inputs an instruction of selection of a filename-for-user*** *under operation of said OS program* ***from the user*** *who is logged in*,

(Col. 3, lines 57-60, “…computer-based file server, operates under control of a UNIX operating system 105...) (Col. 6, lines 32-40, “After the MS-DOS user receives the list of

MS-DOS file name aliases, he/she or it (if a programmatic user) then **selects the desired file name** alias and enters a file call using that file name alias or alternate…").

- ***references the filename correspondence information for the user*** *who is logged in* ***from inside the correspondence information storage unit (80),***

(Col. 9, lines 36-43, "With reference to FIGS. 7 and 8, we describe a lookup strategy which uses a predetermined algorithm (**file name mapping function**) to …")

- ***executes a filename conversion process of converting the selected filename-for-user to a filename-for-storage based on the correspondence information referenced, and provides, to the spread processing unit, an instruction to spread data in a file with the filename-for-storage resulting from the conversion.***

(Col. 7, lines 1-9, "The initial access to a **file** is by its path name, as in the open, chdir (change directory), or link system calls. Because the kernel 130 works internally with vnodes rather than with path names, **it converts the path names to vnodes to access files**. An algorithm of the UNIX system kernel parses the path name one component at a time, **converting each component** into a vnode based on its name and the directory being searched, and eventually returns the vnode of the input path name.").

However Bauer does not disclose:

recognizing a user who is logged in at the present time

from the user who is logged in

an instruction to perform storage using the filename-for-storage;

information for the logged-in user from inside

who is logged in

from the logged-in user

information for the logged-in user

On the other hand Clark discloses the above underlined claimed features as follows:

(Par [0637], “In one embodiment, the synchronization service does not provide its own ... This utility makes it very easy to configure the **Windows Scheduler to run synchronization either on schedule or in response to events such as user logon or logoff.”).**

Neither Bauer nor Clark discloses:

an instruction to perform storage using the filename-for-storage

On the other hand Luke discloses above underlined claimed feature as follow:

(Par [0007], “…An Information Component may be any block of data **or a set of executable instructions comprising an identifiable, storable entity**, but is typically a sub-section of a **file**. …”)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark and Luke into the method of Bauer to take advantage of executing a specific procedure after logging off. The modification would have been obvious because one of the ordinary skills in the art would implement this to keep a system updated depending on user changes throughout logging in and out from the apparatus.

As per Claim 2, the rejection of Claim 1 is incorporated and Bauer further discloses:

> ***- wherein: the storage control unit (35) executes the filename conversion process by generating a filename-for-storage that contains at least algorithmically random codes.***

(Col. 9, lines 44-58, “One might implement "format-dependent matching" **algorithms f**or each type of client operating system that a file server supports. **These format-dependent matching algorithms** would enable "on-the-fly" **conversions** of 1) the format of a base name received in a user program request to the format of the standard file name or 2) **the format of the standard file name to the format of a base name** received in a user program request. ...”) and (Col. 1, lines 19-21, “The operating systems of computers require that file names meet certain constraints. A common constraint is to limit the maximum **number of characters in a file name**.”).

As per Claim 3, the rejection of Claim 1 is incorporated and Bauer further discloses:

- ***wherein: the correspondence information storage unit (80) stores a correspondence table, indicating a correspondence between a filename-for-user and a filename-for-storage, as the filename correspondence information.***

(Col. 2, lines 36-49, “a computer-based file system enables access to any of a plurality of previously-**stored data files stored in a storing means**…The system then **accesses the storage device** and checks file names therein which utilize the identified file name format to locate a data file having a file name which is the same as said base name.") and (Col. 9, lines 36-43, "With reference to FIGS. 7 and 8, we describe a lookup strategy which uses a predetermined algorithm (**file name mapping function**) to determine the DOS **base name equivalents** of the UNIX file names 812, 814 **and 816 of directory 800**. FIG. 9 illustrates a list of **typical algorithms that might be used to map the default (standard) file names** (e.g., 811, 813, 815) to the selected alternate name type.") and (Col. 5, lines 22-26, "When a process creates a new file, the file system driver 132 assigns it an unused inode. Inodes are stored in a section 223 of the physical file system 220, as will be described shortly, but the file system driver 132 reads them **into an in- core-memory inode table when manipulating files**.").

As per Claim 6, the rejection of Claim 1 is incorporated and Bauer further discloses:

> ***- wherein: the storage control unit (35) executes the filename conversion process on the entirety of a filename, including an extension portion.***

(Col. 9, lines 43-58, "…conversions of 1) the format of a base name **received in a user program request to the format of the standard file name** or 2) the format of the standard **file name** to the format of a base name received in a user program request…") and (Col. 1, lines 38-58, "It would be …extension format, and includes an illegal character (a blank space).").

As per Claim 7, the rejection of Claim 1 is incorporated and Bauer further discloses:

- ***wherein: the storage control unit (35) executes a conversion process of converting not only a filename but also contents of timestamps or other***

> ***attribute information that are stored along with a file, the correspondence information storage unit (80) executes a process of storing, as the filename correspondence information, not just a correspondence of filenames but also a correspondence of attribute information before and after the conversion process; and the spread control unit (45) executes a process of restoring converted attribute information based on the correspondence of attribute information before and after the conversion process.***

(Col. 2, lines 36-49, "a computer-based file system enables access to any of a plurality of previously-**stored data files stored in a storing means**…The system then **accesses the storage device** and checks file names therein which utilize the identified file name format to locate a data file having a file name which is the same as said base name.") and (Col. 4, lines 14-31, "The file system driver 132 manages files, allocating file space, controlling access to files, and retrieving data for users. **Processes interact with the file system driver** 132 via a specific set of system calls, such as open (to open a file for reading or writing), close, read, write, stat (query the **attributes of a file**), chown (change the **record of who owns the file**) and chmod (change the access permissions of a file). "attributes" include "timestamps".

As per Claim 9, the rejection of Claim 1 is incorporated and Bauer further does not disclose:

- ***wherein: the correspondence information storage unit (80) is arranged from a portable information storage medium that can be freely attached to and detached from a main body of the computer system.***

On the other hand Luke discloses the above underlined claimed features as follow: (Par [0011], "… data storage repository, and for this reason the invention is particularly advantageous for **portable data processing systems and devices** which have a relatively small storage capacity (**for example, laptop computers, PDAs and other small processing devices**). ...").

As per Claim 10, the rejection of Claim 1 is incorporated and Bauer further discloses:

- ***wherein: functions of the storage control unit (35), the spread control unit (45), and the presentation control unit (65) incorporate a dedicated application program in a computer and the storage control unit (35), the spread control unit (45), and the presentation control unit (65) are made to operate only when said program is started up.***

(Col. 2, lines 8-11, "In accordance with the present invention, I have solved the above-described problems by enabling a user **program** file access request to identify the file name format ...") and (Col. 4, lines 14-31, "The file system driver 132 manages files, **allocating file space**, controlling access to files, and retrieving data for users. **Processes interact with the file system driver**...) and (Abstract, "A **computer-based file system** enables user access to any of a plurality of previously-**stored data files**, each file being identified by at least two file names formatted using different file name formats…") and (Col. 4 and 5, lines 61-67 and 1, respectively, "With joint reference to FIGS. 1, 2 and 3 we describe an overview of a file system. … (/). The internal representation of a file is given by an inode, 200, which contains a description of the disk layout of the file data and other information such as the file owner, access permissions, and access times."), the "presentation control unit" being the "internal representation".) and (Col. 3, lines 58-65, "the computer-based file server, **operates under control** of a UNIX operating system 105, shown using a high-level architecture layer diagram. The layer diagram includes a **user level** 120, a kernel level 130, and a hardware level 140. ..."). "user level" activates the desired program.

As per Claim 12, the rejection of Claim 1 is incorporated and Bauer further discloses:

> ***- A computer readable storage medium including a program making a computer function as the storage control unit (35), the spread control unit (45), and the presentation control unit (65) of the computer system according to Claim 1.***

(Abstract, "A **computer-based file system** enables user access to any of a plurality of previously-**stored data files**, each file being identified by at least two file names formatted using different file name formats…").

However Bauer does not discloses

> *a computer readable storage medium including*

On the other hand Luke discloses the above underlined limitation as follow:

(Abstract and Claim 14, “A computer program product comprising program code recorded on a computer-readable recording medium, the program code including instructions for controlling the operation of a data processing apparatus…”)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark and Luke into the method of Bauer to take advantage of executing a specific procedure and program. The modification would have been obvious because one of the ordinary skills in the art would implement this to keep a system performing according to the required instructions.

11. Claims 4 and 5 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Bauer et al (US Patent No. 5,627,996)**, in view of **Clark et al. (2005/0050054 A1)** and **Luke et al. (US Publication No. 2002/0116402)** and **Makita et al (2004/0123111 A1).**

As per Claim 4, the rejection of Claim 1 is incorporated and Bauer further discloses:

- ***wherein: the storage control unit (35) executes the filename conversion process from a filename-for-user to a filename-for-storage by generating a filename-for-storage <u>based on an algorithm for reversible conversion</u>.***
- (Col. 9, lines 44-58, “One might implement "format-dependent matching" algorithms for each type of client operating system that a file server supports. **These format-dependent matching algorithms** would enable "on-the-fly" **conversions** of 1) the format of a base name received in a user program

request to the format of the standard file name or 2) the format of the standard file name to the format of a base name received in a user program request. ...").

- On the other hand Bauer does not disclose:

<u>based on an algorithm for reversible conversion</u>

However Makita discloses the above underlined claimed features as follows:

(Par [0058], "If it is confirmed that there is no alteration, the TC server F (5F) calculates a hash value 613 by applying the predefined hash **algorithm to the format reverse-conversion** program 605, and decrypts…").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark, Luke, and Makita into the method of Bauer to take advantage of executing reversible conversion based on algorithms. The modification would have been obvious because one of the ordinary skills in the art would implement this to maintain a balanced system for any inputted data and provide a recoverable file from any of user and system naming conversion.

As per Claim 5, the rejection of Claim 4 is incorporated and Bauer further discloses:

- ***wherein: the correspondence information storage unit (80) stores information, <u>indicating the reversible conversion algorithm</u> used in the filename conversion process, as the filename correspondence information.***

(Col. 2, lines 36-49, “a computer-based file system enables access to any of a plurality of previously-**stored data files stored in a storing means**…The system then **accesses the storage device** and checks file names therein which utilize the identified file name format to locate a data file having a file name which is the same as said base name.”) and (Col. 9, lines 44-58, “One might implement "format-dependent matching" algorithms for each type of client operating system that a file server supports. **These format-dependent matching algorithms** would enable "on-the-fly" **conversions** of 1) the format of a base name received in a user program request to the format of the standard file name or 2) the format of the standard file name to the format of a base name received in a user program request. ...”).

- On the other hand Bauer does not disclose:

based on an algorithm for reversible conversion

However Makita discloses above underlined claimed features as follows:

(Par [0058], “If it is confirmed that there is no alteration, the TC server F (5F) calculates a hash value 613 by applying the predefined hash **algorithm to the format reverse-conversion** program 605, and decrypts…”).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark, Luke, and Makita into the method of Bauer to take advantage of executing reversible conversion based on algorithms. The modification would have been obvious because one of the ordinary skills in the art would implement this to maintain a balanced system for any inputted data and provide a recoverable file from any of user and system naming conversion.

12. Claim 8 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Bauer et al (US Patent No. 5,627,996)**, in view of **Clark et al. (2005/0050054 A1)** and **Luke et al. (US Publication No. 2002/0116402) and Reference AL (IDS).**

As per Claim 8, the rejection of Claim 1 is incorporated and Bauer further discloses:

- ***wherein: the correspondence information storage unit (80)*** ***<u>executes a process of encrypting and then storing the filename correspondence information and executes, upon receiving a reference of the stored filename correspondence information, a process of decrypting and then presenting the information to be referenced.</u>***

(Col. 2, lines 36-49, "a computer-based file system enables access to any of a plurality of previously-**stored data files stored in a storing means**…The system then **accesses the storage device** and checks file names therein which utilize the identified file name format to locate a data file having a file name which is the same as said base name.").

However, Bauer does not disclose:

- ***<u>executes a process of encrypting and then storing the filename correspondence information and executes, upon receiving a reference of the stored filename correspondence information, a process of decrypting and then presenting the information to be referenced.</u>***

On the other hand Foreign Document number 2003-288247, reference AL, hereinafter AL, discloses the underlined claimed features as follow:

(Following citation was taken from machine translation document from mentioned reference).

(Par [0017], "encryption and decription algorithm") and (Par [0018], "contents storing part…file storage location…) and (Par [0019], "...filename, .the time that the file uploaded..Contents ID...").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark, Luke, and AL into the method of Bauer to take benefit from the process of encrypting and decrypting. The modification would have been obvious because one of the ordinary skills in the art would implement this to provide the most effective way to achieve data security.

13. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Bauer et al (US Patent No. 5,627,996)**, in view of **Clark et al. (2005/0050054 A1)** and **Luke et al. (US Publication No. 2002/0116402)** and **Ditlow et al (6,788,302 A1).**

As per Claim 11, the rejection of Claim 1 is incorporated and Bauer further discloses:

- wherein: the file storage unit (50) is arranged from a <u>plurality of data storage devices (51, 52, 53) that are configured distributively, the storage processing unit (30A) is provided with a function of dividing a file to be stored into a plurality of partition files and storing the respective partition</u>

files in different data storage devices (51, 52, 53), the spread processing unit (40A) is provided with a function of synthesizing and thereby restoring the plurality of partition files, respectively stored in different data storage devices (51, 52, 53), to an original file and then spreading the original file on the memory, and the correspondence information storage unit (80A) is provided with a function of storing filename correspondence information indicating a correspondence of a "single filename-for-user" to a "plurality of filenames-for-storage," used as respective filenames of the partition files.

(Col. 4, lines 14-31, “The file system driver 132 manages files, **allocating file space**, controlling access to files, and retrieving data for users. **Processes interact with the file system driver...)**

On the other hand Ditlow discloses above underlined claimed features as follows:

,

(Abstract, “The present invention **divides a large graphics file** into smaller "frames" of graphics files. The **division process** is preferably load balanced amongst **any number of processors**. This allows **many processors** to be used in parallel to divide the large graphics file and to then process the smaller output frames. Additionally, the load balancing is performed in such a manner that only portions of the graphics file need be loaded by any one processor. This saves memory and computational requirements. Preferably, the graphics **file is divided** in a three-dimensional manner, such that any one processor will be assigned one three-dimensional block or volume of the graphics file. ...") and (Abstract and Claim 1, "…storing a data density value…").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark, Luke and Ditlow into the method of Bauer to take advantage of dividing the files using partitioning. The modification would have been obvious because one of the ordinary skills in the art would implement this to keep a system updated depending on user changes throughout logging in and out from the apparatus and divide the files so they spread evenly on its destination.

14. Claim 13 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Bauer et al (US Patent No. 5,627,996)**, in view of **Clark et al. (2005/0050054 A1)**

As per Claim 13, Bauer discloses:

- A file storage/read-out method that makes a computer system execute (Title, Method and apparatus for accessing the same computer file using different file name formats").

- ***a storage process of storing data as a file with a predetermined filename into a file storage unit (50) and a readout process of reading out data in a file stored in the file storage unit, said method making the computer***

system execute under operation of a particular OS program, in the storage process:

(Abstract, “A **computer-based file system** enables user access to any of a plurality of previously-**stored data files**, each file being identified by at least two file names formatted using different file name formats…”) and (Col. 3, lines 57-60, “…computer-based file server, **operates under control of a UNIX operating system** 105...) and (Col. 4, lines 15-31, “The system call interface 131…The file system driver 132 manages files, allocating file space, controlling access to files, and retrieving data for users. **Processes interact with the file system driver** 132 via a specific set of system calls, **such a**s open (to open a file for **reading or writing**), close, **read, write**, stat (query the attributes of a file)…”).
(Col. 4, lines 47-53, “...process control subsystem 133 interact when loading a file into memory for execution...”).

a storing user recognition step of recognizing a user who is logged in at the present time; a storage filename input step of inputting a filename-for-user to be assigned to a file to be stored;

(Col. 11, lines 12-15, “More generally in response to a **client computer user input, a user program may specify via a system call or other mechanism the file name format for subsequent purported file names entered by the user.**”) and (Col. 8, lines 19-22, "In step 501 the requester's **execute permission** in the current directory is checked in the standard way. If permission does not exist, an access error message is returned to the user in step 502”). "execute permission" inherently has a "user recognition unit".

a filename conversion step of converting the filename-for-user to a filename-for-storage based on a predetermined algorithm; a correspondence information storage step of storing information, indicating a correspondence between the filename-for-user and the filename-for-storage, as filename correspondence information for <u>a user who is logged in</u>;

(Col. 7, lines 1-9, “The initial access to a **file** is by its path name, as in the open, chdir (change directory), or link system calls. Because the kernel 130 works internally with vnodes rather than with path names, **it converts the path names to vnodes to access files**. An algorithm of the UNIX system kernel parses the path name one component at a time, **converting each component** into a vnode based on its name and the directory being searched, and eventually returns the vnode of the input path name.”) and (Col. 9, lines 36-43, “With reference to FIGS. 7 and 8, we describe a lookup strategy which uses a predetermined algorithm (**file name mapping function**) to determine the DOS **base name equivalents** of the UNIX file names 812, 814 **and 816 of directory 800**. FIG. 9 illustrates a list of **typical algorithms that might be used to map the default (standard) file names** (e.g., 811, 813, 815) to the selected alternate name type.”).

- and a file storage step of storing the file to be stored into the file storage unit under the filename-for-storage; and in the readout process: a reading <u>user recognition step of recognizing a user who is logged in at the present</u>

> ***time; a readout filename input step of inputting a filename-for-user for specifying a file to be read out, a filename referencing step of referencing a filename-for-storage, corresponding to the input filename-for-user, based on the filename correspondence information for a user who is logged in; and a file readout step of reading out a file stored in the file storage unit under the filename-for-storage, obtained by the referencing, as the file to be read out.***

(Col. 11, lines 12-15, "More generally in response to **a client computer user input**, a **user program may specify via a system call** or other mechanism the file name format for subsequent purported **file names entered by the user**. Thus, for example, such a system call may identify the filename format to be utilized on all file name accesses by the user during a predefined period of time (e.g., a session) or until the user re-specifies the original, or another, format again. Such an arrangement enables the user to change the format on a session basis rather than on an individual file name access basis")

However Bauer does not disclose:

> ***who is logged in at the present time***
>
> ***a user who is logged in;***
>
> ***user recognition step of recognizing a user who is logged in at the present time***
>
> ***a user who is logged in***

On the other hand Clark discloses the claimed features as follow:

(Par [0637], “In one embodiment, the synchronization service does not provide its own ... This utility makes it very easy to configure the **Windows Scheduler to run synchronization either on schedule or in response to events such as user logon or logoff.”),** inherently the user is “logged” as claimed.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of invention was made to incorporate the teachings of Clark into the method of Bauer to take advantage of executing a specific procedure after logging off. The modification would have been obvious because one of the ordinary skills in the art would implement this to keep a system updated depending on user changes throughout logging in and out from the system.

As per Claim 14, the rejection of Claim 13 is incorporated and further Bauer disclose:

- wherein: the readout filename input step is executed by a method whereby filenames-for-user corresponding to respective files stored in the file storage unit are referenced and displayed in a list based on the filename correspondence information and a user is made to select a specific filename from among the filenames displayed in the list.

(Col. 10, lines 48-52, “Obviously, if the computed file names were already stored as altname1 831, 833 and 835 in entries 801, 803 and 805, respectively, of directory 800, the server could merely output **this list of file names directly to the client**.”)

(Col. 10, lines 558-66, “By first obtaining a list of alternate file names of directory 800, or the computed file names, the client computer user **can select and then access the desired file.** The server, knowing that the client was at a DOS machine…”).

Conclusion

15. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

16. Any inquiry concerning this communication or earlier communications from the examiner should be directed to ANGELICA RUIZ whose telephone number is (571)270-3158. The examiner can normally be reached on 8:00 a.m. to 4:30 p.m., ET.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Mohammad Ali can be reached on (571) 272-4105. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Angelica Ruiz
Examiner
Art Unit 2169
/J. M. C./

/Mohammad Ali/
Supervisory Patent Examiner, Art Unit 2169